自 然 言 語 処 理　149－17
音声言語情報処理　41－8
（２００２．５．２４）

シナリオ記述を状況に依存して実行する対話エージェントのアーキテクチャ

高田 司郎[†]　山口 毅[‡]　河原 達也[††]　間瀬 健二[†]

[†] (株)ATR メディア情報科学研究所　[‡] (株)ATR 開発センタ
[††] 京都大学大学院情報学研究科

概要

我々は，人間型ロボット，センサーぬいぐるみ，ウエアラブル，ユビキタス環境，実世界指向エージェントなどの形態として協創パートナーを実現することで，操作性の良いインタラクションメディアの提案を目指している．本稿では，協創パートナーとして，状況に依存してシナリオ記述を実行することで，人間と協調的にコミュニケーションを行う対話エージェントのアーキテクチャ，適用事例，および，今後の課題について述べる．

Voice-interactive Agent Architecture Performing Scenario According to Situations

Shiro Takata,[†]　Tsuyoshi Yamaguchi,[‡]　Tatsuya Kawahara[††] and　Kenji Mase[†]

[†] ATR Media Information Science Laboratories
[‡] ATR Technology Liaison Center
[††] Graguate School of Informatics, Kyoto University

Abstract

The aim of our research is to propose operational interaction media realied by co-creative partners in the form of humanoid robots, sensor stuffed doll, wearables, ubiquitous systems and real-world agents. In this paper, a voice-interactive agent architecture is proposed. The agent can communicate with humans cooperatively as a co-creative partner according to situations by performing scenario. Then, an application case and future problems of this architecture are described.

1　はじめに

我々は，実世界コンピューティングをめざして，ロボット，ウエアラブル，ユビキタス環境，実世界指向エージェントなどの人工物を組み合わせた，誰でも，いつも付き添う「人工生涯パートナー」の研究に着手している．このようなパートナーには，広い年齢層の人間に対して，自律的，協調的にインタラクションを創り上げる能力が必要である．そこで，このようなパートナーを「協創パートナー」と呼び，人間との多様なインタラクションがもたらす協調的な活動に着目して，実世界コンピューティングが可能で，操作性のよいメディア技術の研究に取り組んでいる [1, 2].

本稿では，協創パートナーの研究開発の一貫として，人間との日常的，協調的，および，状況に依存したインタラクションをシナリオとして記述して協創パートナーに与えると，そのシナリオを自らのプランとして自律的（意図的）に実行することで，文脈や動的環境などの状況に依存して，人間と協調的にコミュニケーションを行う対話エージェントのアーキテクチャについて述べる．次に，このアーキテクチャの適用事例，および，今後の課題について述べる．

2 動的に変化する状況に依存した行為

我々は，以下のような動的に変化する状況に依存した行為生成を研究課題としている．

(1) 音声韻律，視線，ジェスチャ，合図，その他計測可能な五感情報などのマルチモーダルインタラクションによる局所的な焦点移動に対する照応行為

(2) 話者人数が変化した場合の発話行為

(3) 会話進行障害からの修復連鎖行為

(4) 文脈に依存した行為（主に発話行為）

協創パートナーは，まず，実世界指向に立ち，個々の様相を分担するマルチエージェントから構成されるマルチモーダルインタフェースから部分的な外界状況を取得し，その状況がそのプラン遂行条件として記述されたシナリオがあれば，そのシナリオに沿ったインタラクションを即応的に繰り返して，徐々に，パートナーが置かれた状況（シーン）を理解していく．このようなシナリオを，シーン把握シナリオと呼び，人間との日常的，協調的な振る舞いを決定するために非常に重要なシナリオと考えている．これらシナリオは，大量の五感データのコーパスから分析・抽出できるものと考えている．

そして，シーン把握ができたとする．次に，協創パートナーは，人間間で合理的だと社会学習した行為が記述されたシナリオを実行していくことになる．これらシナリオに記述された合理的なプランは，人間間の社会的相互作用の知恵であり，常識的かつ広大な社会知の範囲の課題と捉えられる．そこで，我々は，合理的行為のシナリオの収集は，以下で扱う適用事例に絞り，シーン把握シナリオの分析・収集を中心課題とする．

以下，このような状況把握を行い，かつ，合理的な行為を行うシナリオを与えることで人間と協調的に，しかも，状況に沿った合理的行為を行う協創パートナーを実現するアーキテクチャを考える．

3 アーキテクチャ

計算機の記憶容量・処理時間など資源は有限であり必要な情報を全て保持したり，今の状態から未来に渡る全ての合理的な行為を推論したりすることは出来ない．また，我々を取り巻く世界は絶えず変化しているため，行為を行う前に，事前に立てた計画の多くが無駄に終わるかもしれない．そこで，我々は，動的に変化する環境下で，状況に応じて，人間または他のエージェントと協調して問題解決を行う合理的エージェントの実現手法として，拡張 BDI アーキテクチャを提案している [3]．今回，前節の課題に対処するための取り組みとして，対話制御の柔軟性の向上，および，マルチモーダルインタフェースの実現方式のカプセル化を目指して，以下のプラン記述，アーキテクチャに拡張した．

3.1 プラン記述

我々は，対話を [質問-返答] や [誘い-承諾] などを基本的な相互行為とする隣接ペアを最小単位としたスタックモデルの談話構造と捉えて，対話プランを記述する．そして，あるシーンの隣接ペアの展開を記述したサブプランを，状況に応じて（未来指向的意図形成条件）意図として形成した後，その時が来れば（現在指向的意図遂行条件）意図スタックに積んで，意図的に実行する．そこで，このような対話制御を柔軟に記述するために，プランの実行部に prolog 述語の記述を許し，下記のような文法に拡張した．また，サブプランを別途設けた．

```
Body ::= {Statement}+
Statement ::= prolog 述語
  | subplan(プランタイプ名)
  | guard([{[Condition, Action]}+
          {[otherwise, Action]}])
  | if (Condition, then Statement
          {, else Statement})
  | until(Condition, Action)
  | do_until(Action, Condition)
Condition ::= [{prolog 述語}+]
Action ::= [{Statement}+]
```

ただし，{}は省略可，{}+は一回以上の繰り返しを表す．

```
sub_plan(
  Type,             ... プランタイプ名
  [],               ... 環境からのイベント
  [FormCondition,   ... 未来指向的意図形成条件
   PreCondition,    ... 現在指向的意図遂行条件
   AddList,         ... 効果期待（追加）
   DeleteList,      ... 効果期待（削除）
   Body             ... 実行部
  ])
```

3.2 マルチモーダルインタフェース

マルチモーダルインタフェースの実現には，各様相を外界の一つの状況と捉え，そのインタフェースを拡張 BDI アーキテクチャの入出力部に接続する方式を採用する．たとえば，音声インタフェースは，マルチモーダルインタフェースドライバを通して，拡

張 BDI アーキテクチャの入出力部とソケット（または，パイプ）接続している（図 1）．また，画像処理は，その他のエージェント群から構成する．音声認識ソフトは，Julian [4]，音声合成ソフトは，ATR CHATR を使用している．また，音声出力として，録音データを直接出力することもできる．

3.3 音声インタフェース

音声インタフェースは，一つの隣接ペアを単位に，音声出力，Julian の起動および認識結果からキーワード抽出を行い，それら結果を拡張 BDI アーキテクチャの入出力部に通知する．特に，認識率向上のため，その隣接ペアに限定したネットワーク文法と音響モデルを指定して，毎回 Julian を起動し，認識後，終了させている．また，Julian の起動に当たっては，音声出力のサイズを計算し自らの音声出力を入力しないよう工夫している．そこで，下記のような音声認識要求コマンドを使用する．このようなコマンド形式を採用することで，シナリオライターは，音声入出力処理がどのように実現されているかを意識することなく対話プランを記述できる．

```
・音声認識要求コマンド
task=TASK sex=SEX [rsex=RSEX wave=WAVE]
  TASK:= ネットワーク文法名 | skip （文法指定）
  SEX:=  male  | female | gid （音響モデルの指定）
  RSEX:= male  | female（発話の性別指定）
  WAVE:= [wave1,wave2,...]
         | 音声出力テキスト
・認識結果通知
inform(keyword([[KEYWORD,PROB],...]), TASK).
  KEYWORD:= 認識したキーワード文字
  PROB:= 認識したキーワードの確からしさ
  TASK:= 要求メッセージ中の現在の TASK
inform(keyword([[], STATUS]), TASK).
  STATUS:= nokeyword | overfillong | sptimeout
inform(status(STATUS,TASK ), []).
```

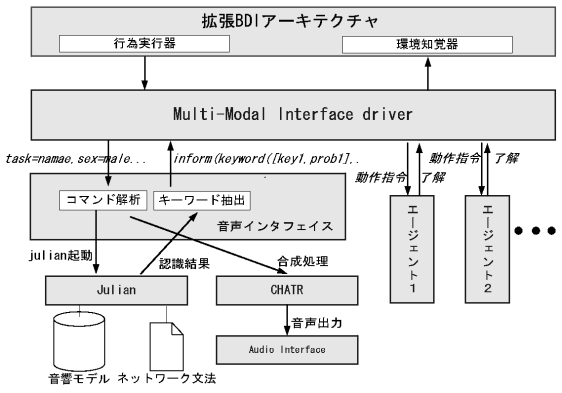

図 1: マルチモーダルインタフェース

```
  STATUS:=  loud | quiet | error
  TASK:= 要求メッセージ中の現在の TASK
```

4 適用事例

我々は，2 節の研究課題 (3)(4) および (1)(2) のテストベッドとして，(a) 子供と日常会話を行う対話エージェント，(b) 利用者と協調して写真を撮る対話エージェントを，それぞれ，開発している．以下，今までの取り組みと今後の課題について述べる．

4.1 子供との日常会話エージェント

4.1.1 日常会話のシナリオ

まず，子供（3 歳から 16 歳が中心）を対象に，日常型ロボット Robovie を用いて WOZ を実施，1 対話 3 分程度の 50 対話例を収集した．その対話例の分析から，名前の質問，年齢の質問，住所の質問，学校の質問，同伴者の質問，来館目的の質問，ブースの質問，将来の夢の質問などのシーンに分類して，シナリオ作りを実施した．各シーンは，システムから利用者への「質問」から始まり，利用者からの「返答」が認識できれば，付加的な「質問」を利用者に発話するなどの隣接ペアを基本とした．認識できないときは，優先的応答として，同意の「ふーん」などを発話する．たとえば，下記の例では，年齢を聞き学年を推定した後，学校の質問を実施している．

```
 S：何歳ですか？
 U：えーっと，6 歳です.
 S：小学 1 年生ですか？
 U：はい.
 S：学校は好きですか？

plan(インタビュー,
 inform(request(StartMessage),[]),
 [[true],[true],[],[],
  [profile_initialize(StartMessage),
   subplan(挨拶),
   subplan(年齢の質問),
   subplan(学校の質問),
      …
   interview_terminate]
 ]).
plan(音声認識割り込み,
 inform(status(Status,Network),[]),
 [[true],[true],[],[],
  [interrupt(Status,Network)]
 ]).
  …
sub_plan(年齢の質問, [],
 [[true], [true],
  [profile(利用者, 年齢 (Age,Eval)),
   profile(利用者, 学年 (Degree))], [],
   [ask_age,
```

```
    guard([[keyword([[],Eval]),
      (Age=[],reply_age([],Eval),exit)],
     [keyword([Age,Eval]),
       reply_age(Age,Eval)]]),
    guard([[keyword(Degree),
       reply_degree(Degree)]])]
]).
sub_plan(学校の質問, [],
 [[(profile(利用者, 年齢(Age,Eval)), Age>=6)],
  [true],
  [profile(利用者,
           学校の好き嫌い(LoveSchool,Eval)),
   profile(利用者, 科目(Subjects))], [],
   [ask_school_love,  % 学校の好き嫌いを聞く.
     guard([[keyword([[], Eval]),
       (LoveSchool = [],
        reply_school_love([], Eval))],
     % 返答のみで終了するか, さらに科目を聞く.
      [keyword([LoveSchool, Eval]),
        if(user_mod(2, 0),
         reply_school_love(LoveSchool, Eval),
         [ask_school_subject(LoveSchool),
           guard([[keyword(Subjects),
            reply_school_subject(Subjects,
             [LoveSchool,Eval])]
      ...
```

4.1.2 文脈に依存した発話行為

シナリオの文脈は，シナリオ分析の結果得た各シーンで使用されるキーワードの集合と捉えられる．そこで，各シーン単位に，下記のような Julian のネットワーク文法と辞書を作成し音声認識結果からキーワードを抽出する方式とした．キーワード抽出には，キーワード以外は，FILLER として @ をマークし，図 1 の音声インタフェースで除いた．たとえば，下記の学校の質問のシーンでは,「学校は好きですか?」という質問文，そのシーンに使用する文法と辞書を指定して,「おもしろい」「まあ」「おもしろくない」などのキーワードを guard で受け，それまでの認識結果などを参照しながら，そのキーワードに該当する返答文を生成するといったシナリオを記述することで，学校の質問という文脈における発話行為を行う．

```
# 学校の質問用文法
S: NS_B HENJI_S NS_E
HENJI_S: FILLER_S NOISE HENJI FILLER_S
HENJI_S: FILLER_S NOISE HENJI
HENJI_S: HENJI FILLER_S
HENJI_S: FILLER_S
HENJI_S: HENJI
FILLER_S: FILLER
FILLER_S: FILLER_S FILLER

# 学校の質問用辞書
#キーワード
%HENJI
おもしろい      h a i
おもしろい      h a: i
...
おもしろい      d a i s u k i
まあ            m a a
...
おもしろくない  d a i k i r a i
おもしろくない  ts u m a r a n a i
おもしろくない  ts u m a r a N
おもしろくない  ts u m a N n a i
...
%FILLER
@すごく         s u g o k u
@まったく       m a q t a k u
@学生ちゃう     g a k u s e i ch a u
@学生とちがう   g a k u s e i t o ch i g a u
@学校いってへん g a q k o u i q t e h e N
...
@ありがと       a r i g a t o
@ばいばー       b a i b a:
@きょーわ       ky o: w
```

4.1.3 話題の修復

小学校に，まだ，行っていない子供に対して，学校の好き嫌いの質問をすると，ほとんどの子供は，まず「学校になんか行ってないよ！」という回答をする．この回答に対応せず「学校で何が一番好き？」とかの質問を続けると,「学校になんか行っていないと言ってるやろ！ この馬鹿（関東のアホの意味）!」と怒り出す．仲間と一緒に来ている場合はなおさらである．現在は，このような状況を回避するために,「学校の質問」の未来指向的意図形成条件に「子供から聞き取った年齢が，6 歳未満であれば，学校の質問を意図として形成しないようにしている．

本来，このような状況から修復するためには，(1) 学校には行っていないという発話を理解して謝るようなシナリオを用意しておく．(2) 子供の発話内容の理解ではなく，音韻などからこの質問には対応できないという子供の意図を把握して，全く別の Yes/No 質問に切り替えるようなシナリオを用意しておくことなどが考えられる．

また，話者の声が小さすぎたり大きすぎたりする状況では，音声認識はできない．このような状況は,隣接ペアの全ての応答で発生するが，個別に記述して修復する問題ではない．このような状況から修復するために,「音声認識割り込み」プランを用意している．このプランは，意図として形成されると最優先に実行されるように，マルチモーダルインタフェースからの通知 (inform) を，要求 (request) ではなく，状態 (status) として区別している．このプランは，たとえば,「もう少し大きい声で話してね!」と一回だけ発話行為を行い，さらに，この状況を生じさせた音声認識用のネットワーク文法を音声インタフェースに再送することで，シナリオの継続を可能としている．このプランは，シーン把握シナリオの簡単な

事例と考えられる．

4.2 写真を撮る協調エージェント

体験を記録するメディアとして，写真やビデオは重要な位置を占めている．従来は，漠然と録画された後，個人の意図に基づいて編集していた．Photo-Agent は，撮影時に複数の利用者の対話を促進して和やかな環境を作ると共に彼らの意図を理解しつつ，その意図に沿ったレイアウトで仲間との写真やビデオを取ることを開発目標としている．

また，当試作は，2 節の研究課題 (1) マルチモーダルインタラクション (2) 多人数対話の研究を目的として，以下のような課題設定をした，協創パートナー研究のテストベットである：(a) マルチモーダルなセンサーやアクチュエータを駆使した実世界の理解，(b) 体験コーパス [2] から最適なシナリオの検索，(c) 動的に変化する実世界におけるシナリオの演出．

以下，現在の開発状況に触れ，これらの試作から得た課題を次節で述べる．

4.2.1 協調行為

Photo-agent のシステム構成を，図 2 に示す．図 2 の左カメラは，親カメラと小カメラを用いて，左右上下にこれらカメラを動かして利用者の顔追跡を行う [5]．同時にこの相対的な動きと連動して，右カメラを左右上下に動かす．右カメラは，利用者との音声対話を通じて，たとえば,「右に！」「次は，すこしズームして！」などの指示に従って，左カメラからの指示とは独立に，左右上下の移動，および，ズームの操作などを行なう．最初，左右のカメラの向きは，左カメラで最初に検出した眉間がカメラの中央に来る位置に設定する．以後は，左カメラは，眉間追跡により，常にカメラの中央に利用者が居るように動く．右カメラも相対的に移動するが，利用者からの指示を反映した状況を保つ．

また，写真のレイアウトは，右カメラからの映像をテレビ画面に写し出し，利用者に表示する．これらの機能より，利用者が左右上下に動いても，利用者が指示したレイアウトを，右カメラからの映像に保つことができる．

そこで，利用者は，テレビ画面を見ながら音声にて，レイアウトを指示し，要望に沿いそうになれば「フリーズ！」にてテレビ画面を一時的に止めることができる．その画面が気に入れば，キープ指示にで，写真にすることもできる．また，レイアウト指示中に,「撮って！」などの指示で，その都度，写真を撮ることもできる．これら撮られた写真は，ホームページ形式で，即座に，別のディスプレイに表示され，利用者は，それら写真を眺めながら，満足するまで撮り続けることができる．

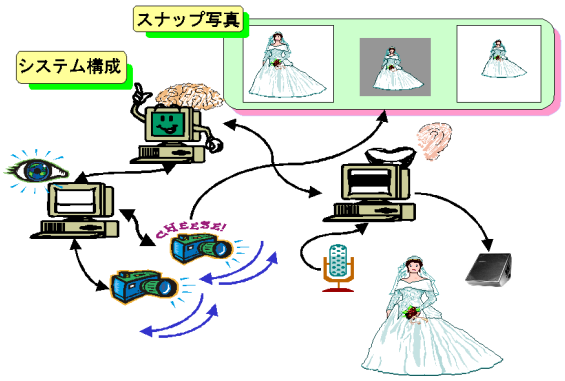

図 2: Photo-Agent のシステム構成図

4.2.2 シナリオ

Photo-Agent の主なシナリオは下記の通りである．

- 眉間抽出技術を用いて，利用者の顔を検出し，以後，顔追跡を継続して行う．
- 利用者の気持を和らげるために，挨拶など日常的な会話を行う．また，この会話を行うことで，音声対話の認識度合いを利用者に肌で感じて貰う．
- 利用者に左右に動くように依頼して，顔追跡結果をディスプレイで見せることで，この技術のデモを行う．
- 実際に利用者の写真を取り，利用者が音声対話で何が指示できるのか理解させる．
- 音声対話を通じて，利用者と協調して写真を撮る．

また，利用者と挨拶を交わし，写真を撮影するシーンのプラン記述例を以下に記述する．

```
plan(写真撮影,
  inform(request(StartMessage),[]),
  [[true], [true], [], [],
   [profile_initialize(StartMessage),
    subplan(御用伺い),
    subplan(写真撮影),
    interview_terminate]]).
sub_plan(御用伺い, [],
```

```
  [[bel(profile(写真マン, 体調(良好)))],
   [true],[],[],
   [subplan(挨拶),
    subplan(名前の質問),
    subplan(撮影目的の質問)]]).
    ....

sub_plan(撮影,[],
  [[true],[true],
   [来館者(撮影回数(M)),
    写真マン(前回行為(撮影))],
   [写真マン(前回行為(_))],
   [subplan(撮影サイン),
    (user_mod( 5, Index ),
     member([Index, Message],
      [[0,ぱちり],[1,かしゃっ],[2,がしゃっ],
       [3, ぱち],[4, かしゃ]]),
     sound_and_action(Message , skip, 撮影),
     retract(belief(来館者(撮影回数(N)))),
     M is N + 1)]]).
sub_plan(カメラ操作, [],
  [[true],[true],[],[],
   [(user_mod( 7, Index ),
     member([Index, Message],
      [[0,いい?],[1,どう?],[2,これでどう?],[3,
まだ?],
       [4,つぎは?],[5,とる?],[6,きーぷ?]]),
     sound_and_action( Message, operate, 依
頼)),
     guard([[keyword([[],nokeyword]),
          sound_and_action(いろいろ言ってみてね,
                           skip,依頼)],
       [keyword([[],sptimeout]),
          sound_and_action(じゃあ, ぱちり,skip,
撮る)],
       [keyword([[], Error]),
          julian_error( Error )],
       [keyword([Operation,_]),
          subplan(Operation)]])]]).
```

当試作システムの評価として,(1)自分の写真を音声指示にて撮るという習慣がないため，何をしているのか分からない利用者が多く見られた.(2)写真屋と違って気楽にレイアウトを見ながら自分の写真が撮れるため，比較的良い表情の写真が撮れていた．などが挙げられる.

5 今後の課題

以上の適用事例などを通じて，以下のような今後の課題が挙げられる.

- デバッガ：拡張 BDI アーキテクチャの動きを理解していないと対話プランのデバッグができないため信念，ゴール，意図などの状態の表示などデバッガが必要である.
- シナリオの簡易記述ツール：表や XML などを用いて計算機の専門家でないシナリオライターと共同作業が行える環境が必要である.
- マルチモーダルインタフェースドライバの汎用化：特に，通過経路指定など，様相に依存しない言語化が必要である.
- 顔向きの同定：画像処理から利用者に正面を向くような発話行為を対話処理に依頼する．対話処理は，何気ない会話を開始して利用者の顔の向きを変化させる．画像処理は，その変化から顔向きに関する認識率を上げることができるかどうか？
- 人数の同定：画像処理で画像解析した対面人数を対話処理に送る．対話処理は，その人数を確かめるための何気ない問い合わせを実施後，結果を画像処理に返す．画像処理は，その情報を基に，人数の認識率を上げることができるかどうか？
- 多人数対話：対話処理は，画像処理からの人数・位置情報を基に多人数対話を行うことができるかどうか？ たとえば，対面から居なくなったという情報を貰ったときなど，そのシーンに応じた多人数対話を行うことができるであろうか？

謝辞 本研究の一部は通信・放送機構の研究委託により実施したものである.

参考文献

[1] 萩田紀博: ダイバシティ・メディアとしての体験 Web 構想, 情報処理学会全国大会, Vol. 4, pp. 411–414 (2002).

[2] 間瀬健二, 角康之, 萩田紀博: 体験 Web における情報処理基盤としての協創パートナーとインタラクション・コーパスの提案, 情報処理学会全国大会, Vol. 4, pp. 551–552 (2002).

[3] 高田司郎, 五十嵐新女, 新出尚之, 榎本美香, 間瀬健二, 中津良平: マルチエージェント環境において意図的に言語行為を遂行する合理的エージェントの基本設計, 電子情報通信学会論文誌, Vol. J84-D-I, No. 8, pp. 1191–1201 (2001).

[4] 河原達也, 住吉貴志, 李晃伸, 武田一哉, 三村正人, 伊藤彰則, 伊藤克亘, 鹿野清宏: 連続音声認識コンソーシアム 2000 年度版ソフトウェアの概要と評価, SLP-38-6 (2001). http://www.lang.astem.or.jp/CSRC/.

[5] 川戸慎二郎, 鉄谷信二: アイカメラへの目位置出力を目的とした目の検出と追跡, 信学技報, PRMU2001-153, pp. 1–6 (2001).